Sparse Domain Adaptation in Projection Spaces based on Good Similarity Functions

Emilie Morvant, Amaury Habrard, Stéphane Ayache

Sparse Domain Adaptation in Projection Spaces based on Good Similarity Functions

Emilie Morvant*, Amaury Habrard† and Stéphane Ayache*
*Aix-Marseille Univ, LIF-QARMA, CNRS UMR 6166, F-13013, Marseille, France
†University of St-Etienne, Laboratoire Hubert Curien, CNRS UMR 5516, F-42000, St-Etienne, France
{emilie.morvant,stephane.ayache}@lif.univ-mrs.fr, amaury.habrard@univ-st-etienne.fr

***Abstract*—We address the problem of domain adaptation for binary classification which arises when the distributions generating the source learning data and target test data are somewhat different. We consider the challenging case where no target labeled data is available. From a theoretical standpoint, a classifier has better generalization guarantees when the two domain marginal distributions are close. We study a new direction based on a recent framework of Balcan et al. allowing to learn linear classifiers in an explicit projection space based on similarity functions that may be not symmetric and not positive semi-definite. We propose a general method for learning a good classifier on target data with generalization guarantees and we improve its efficiency thanks to an iterative procedure by reweighting the similarity function - compatible with Balcan et al. framework - to move closer the two distributions in a new projection space. Hyperparameters and reweighting quality are controlled by a reverse validation procedure. Our approach is based on a linear programming formulation and shows good adaptation performances with very sparse models. We evaluate it on a synthetic problem and on real image annotation task.**

***Keywords*-Machine Learning; Transfer Learning; Domain Adaptation; Binary Classification; Similarity Functions**

I. Introduction

In machine learning, many approaches for learning binary classifiers are built under the assumption that learning data are representative of test data. While this assumption can be relevant for some tasks, it is not always true for every applications. To overcome this drawback, some transfer learning methods [1] have been proposed to adapt a model from a source domain to a target one. In this paper, we address the problem of domain adaptation (DA) where the test data are supposed to be drawn according to a distribution - the *target domain* - different from the one used for generating learning data - the *source domain* [2]. DA is an important issue for the efficient application of machine learning methods and many approaches have been proposed in the literature. While some of them proposed to use a few labeled data from the target domain [3], [4], [5], [6], we consider the more challenging problem where no target labeled data is available.

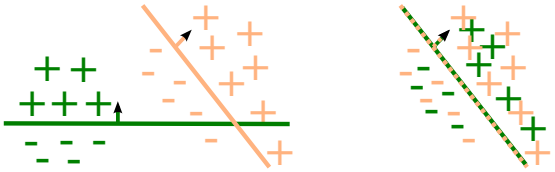
Some theoretical DA frameworks [3], [7] indicate that a classifier learned only from the labeled source data can perform well on the target data if the source and target marginal distributions are relatively close, under the assumption that the two domains are related. This suggests a natural approach for a successful DA: move closer the source and target distributions while keeping a good classifier on the source domain. Following this idea, several methods - based on different hypothesis or discrepancy measures - have been proposed for reweighting the learning data [7], [8], [9]. Bruzzone *et al.* [10] have designed a SVM-based procedure that iteratively removes source labeled points and adds self labeled target points to the learning sample. Another idea consists in finding a common relevant feature space where the two distributions are close [3]. However, this principle relies mainly on heuristics specific to particular tasks.

In this article, we propose a new DA approach for binary classification, based on a recent framework of Balcan *et al.* [11], [12] allowing to learn in an explicit projection space defined by *good similarity functions* that may be not symmetric nor positive semi-definite (PSD) *i.e.* that generalize kernel functions. They show that it is possible to learn a good linear classifier in a space defined by similarities to some relevant landmark examples. These landmarks offer a natural set of features to transfer. Our idea is to automatically modify this projection space for moving closer source and target points, leading to a good adaptation on the target domain. For this purpose, we present a general approach based on a regularizer focusing on landmark points close to both source and target examples. We formulate it in a 1-norm regularized linear program leading naturally to very sparse models. Our approach is then more flexible than SVM-based methods. Moreover, we propose an iterative process based on the absence of PSD and symmetric requirements to improve the tractability of the method. We evaluate it on a synthetic problem and on real image annotation corpora.

The paper is organized as follows. Section II introduces a classical DA theory [3]. Section III deals with the framework of Balcan *et al.* [11]. Our approach is presented in Section IV and its iterative enhancement in Section V. Finally, the algorithm is experimentally evaluated in Section VI.

II. Domain Adaptation

Let $X \subseteq \mathbb{R}^d$ be the input space of dimension d and $Y = \{-1, +1\}$ the label set. A domain is defined as a probability distribution over $X \times Y$. In a DA framework, we have a *source domain* represented by a distribution P_S and a *target domain* represented by a somewhat different distribution P_T. D_S and D_T are the respective marginal distributions over X. A learning algorithm is provided with a *Labeled Source sample* $LS = \{(\mathbf{x}_i, y_i)\}_{i=1}^{d_l}$ drawn *i.i.d.*



(a) A high domain distance: The samples are easily separable, the classifier learned from LS performs badly on TS.

(b) A low domain distance: The classifier learned from LS performs well on the two domains.

Figure 1. The intuition behind Theorem 1. The source points (LS) are in (dark) green (pos.+, neg.−), the target points (TS) are in (light) orange.

from P_S, and an *unlabeled Target Sample* $TS = \{\mathbf{x}_j\}_{j=1}^{d_t}$ drawn *i.i.d.* from D_T. Let $h : X \to Y$ be an hypothesis function which is a binary classifier. The expected error of h over the source domain P_S is the probability that h commits an error: $\mathrm{err}_S(h) = \mathbb{E}_{(\mathbf{x},y)\sim P_S} L_{01}\big(h,(\mathbf{x},y)\big)$, where $L_{01}(h,(\mathbf{x},y)) = 1$ if $h(\mathbf{x}) \neq y$ and zero otherwise, corresponding to the 0-1 *loss function*. The target domain error err_T over P_T is defined in a similar way, $\hat{\mathrm{err}}_S$ and $\hat{\mathrm{err}}_T$ are the empirical errors. An hypothesis class $\mathcal{H}$ is a set of hypothesis from X to Y. We now review the theoretical framework of DA based on Ben-David *et al.* [3], where they give an upper bound for the target domain expected error.

Theorem 1 ([3]). *Let $\mathcal{H}$ be a hypothesis class,*

$$\forall h \in \mathcal{H},\ \mathrm{err}_T(h) \ \leq\ \mathrm{err}_S(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_T) + \nu,$$

where the $\mathcal{H}\Delta\mathcal{H}$-distance between D_S and D_T is $d_{\mathcal{H}\Delta\mathcal{H}}(D_S,D_T) = 2 \sup_{h,h'\in\mathcal{H}\Delta\mathcal{H}} |Pr_{D_S}[h(\mathbf{x})\neq h'(\mathbf{x})] - Pr_{D_T}[h(\mathbf{x})\neq h'(\mathbf{x})]|$ *with $\mathcal{H}\Delta\mathcal{H} = \{h(\mathbf{x}) \oplus h'(\mathbf{x}) : h,h' \in \mathcal{H}\}$ the symmetric difference hypothesis space of $\mathcal{H}$, and $\nu = \mathrm{err}_S(h^*) + \mathrm{err}_T(h^*)$ with $h^* = \mathrm{argmin}_{h\in\mathcal{H}}(\mathrm{err}_S(h) + \mathrm{err}_T(h))$.*

This bound depends on the source domain expected error, which can be easily minimized by a learning algorithm based on the ERM principle. ν is related to the ideal joint hypothesis over the two domains and can be seen as a quality measure of $\mathcal{H}$ for the considered DA task. If this best hypothesis performs poorly, then it appears hard to obtain a good hypothesis for the target domain. The other key point is $d_{\mathcal{H}\Delta\mathcal{H}}(D_S,D_T)$ the $\mathcal{H}\Delta\mathcal{H}$-distance between the two marginal distributions. Theorem 1 suggests that if they are close, then a low error classifier over the source domain can be a good classifier for the target one. The intuition behind this idea is given on Fig. 1. This measure is actually related to $\mathcal{H}$ by measuring a maximum variation divergence over the set of points on which an hypothesis can commit errors. An interesting point is that when the VC-dimension of $\mathcal{H}$ is finite, $d_{\mathcal{H}\Delta\mathcal{H}}$ can be estimated from finite samples.

Lemma 1 ([3]). *If S and T are unlabeled samples of size m i.i.d. from D_S and D_T respectively, $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S,T)$ is the empirical $\mathcal{H}\Delta\mathcal{H}$-distance and v is the finite VC-dimension of $\mathcal{H}$, then for any $\delta > 0$ with probability at least $1-\delta$,*

$$d_{\mathcal{H}\Delta\mathcal{H}}(D_S,D_T) \leq \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S,T) + 4\sqrt{\frac{2v\log(2m) + \log\frac{2}{\delta}}{m}}.$$

$\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ converges thus to the true $d_{\mathcal{H}\Delta\mathcal{H}}$ with the size of the samples. Consider a labeled sample made of $S \cup T$ where each instance of S is labeled as positive and each one of T as negative, we can estimate directly $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ ($\in [0,2]$) by looking for the best classifier able to separate S from T,

$$\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S,T) = 2\left(1 - \min_{h\in\mathcal{H}\Delta\mathcal{H}} \hat{\mathrm{err}}_{S\cup T}(h)\right), \qquad (1)$$

with $\hat{\mathrm{err}}_{S\cup T}(h) = \frac{1}{m}\left[\sum_{\substack{\mathbf{x}\in S\cup T:\\ h(\mathbf{x})=-1}} \mathbb{1}_{\mathbf{x}\in S} + \sum_{\substack{\mathbf{x}\in S\cup T:\\ h(\mathbf{x})=1}} \mathbb{1}_{\mathbf{x}\in T}\right]$,

where $\mathbb{1}_{\mathbf{x}\in A} = 1$ if $\mathbf{x}\in A$ and zero otherwise. Finding the optimal hyperplane is NP-hard in general. However, an estimation of $d_{\mathcal{H}\Delta\mathcal{H}}(D_S,D_T)$ allows us to have an insight of the distribution distance and thus of the difficulty of the DA task for the class $\mathcal{H}$. Note that, Mansour *et al.* [7] have extended the $\mathcal{H}\Delta\mathcal{H}$-distance to real valued functions and have provided Rademacher generalization bounds.

Following Theorem 1, one solution for a DA algorithm is to look for a data projection space where both the $\mathcal{H}\Delta\mathcal{H}$-distance and the source domain expected error of a classifier are low (see Fig. 1). According to [13], minimizing these two terms appears necessary to ensure a good adaptation.

III. Learning with Good Similarity Functions

In this part, we present the framework proposed by Balcan *et al.* of similarity based binary linear classifiers. A similarity over X is any pairwise function $K : X \times X \to [-1,1]$. Many algorithms use similarity functions, like support vector machines where the similarity needs to be a kernel, *i.e.* symmetric and PSD, to ensure learning in the implicit high dimensional space defined by the kernel. Due to the PSD requirement, considering kernels can be a strong limitation and defining a good kernel is a tricky task in general. Balcan *et al.* [11], [12] consider a rather intuitive notion of a *good similarity function* that overcomes some of these limitations. We review it by beginning with their definition.

Definition 1 ([11]). *A similarity function K is an* **(ϵ,γ,τ)-good similarity function** *for a learning problem P if there exists a (random) indicator function $R(\mathbf{x})$ defining a set of reasonable points such that the following conditions hold:*

(i) A $1-\epsilon$ probability mass of examples $(\mathbf{x},y)$ satisfy

$$\mathbb{E}_{(\mathbf{x}',y')\sim P}\big[yy'K(\mathbf{x},\mathbf{x}')|R(\mathbf{x}')=1\big] \ \geq\ \gamma,$$

(ii) $Pr_{\mathbf{x}'}[R(\mathbf{x}')=1] \ \geq\ \tau.$

From this definition, a large proportion of examples must be on average more similar, with respect to the margin

γ, to the reasonable points of the same class than to the reasonable points of the opposite class *(i)*. Moreover, at least a proportion τ of the examples should be reasonable *(ii)*. Definition 1 includes all valid kernels as well as some non-PSD similarity functions and is thus a generalization of kernels [11], [12]. In general the reasonable points are unknown *a priori*. Therefore, in the following we denote by $R=\{\mathbf{x}'_j\}_{j=1}^{d_u}$ a set of *potential* reasonable points called *landmarks*. Given K an (ϵ,γ,τ)-good similarity function, the conditions of Balcan *et al.* are sufficient to learn a good linear classifier in a ϕ^R-space defined by the mapping function ϕ^R, which projects a point in the explicit space of the similarities to the landmarks such that,

$$\phi^R : \left\{ \begin{array}{ccc} X & \rightarrow & \mathbb{R}^{d_u} \\ \mathbf{x} & \mapsto & \langle K(\mathbf{x},\mathbf{x}'_1),\ldots,K(\mathbf{x},\mathbf{x}'_{d_u})\rangle. \end{array} \right.$$

Let LS a set of d_l labeled points, R a set of - enough - d_u landmarks. Then, with a high probability, the induced distribution in the ϕ^R-space has a low error separator with a margin relative to γ. Thus, one can efficiently find a separator $\boldsymbol{\alpha} \in \mathbb{R}^{d_u}$ by solving the linear problem of (2) (based on the hinge loss presented in [11]).

$$\min_{\boldsymbol{\alpha}=\langle \alpha_1,..,\alpha_{d_u}\rangle} \frac{1}{d_l}\sum_{i=1}^{d_l}\left[1-y_i\sum_{j=1}^{d_u}\alpha_j K(\mathbf{x}_i,\mathbf{x}'_j)\right]_+ + \lambda\|\boldsymbol{\alpha}\|_1, \tag{2}$$

where $[1-z]_+=\max(0,1-z)$ is the hinge loss. The learned linear model is then denoted by $g(\mathbf{x})=\sum_{j=1}^{d_u}\alpha_j K(\mathbf{x},\mathbf{x}'_j)$. This leads to a natural two steps algorithm for learning the classifier: select a random set of potential landmarks and then learn a binary classifier $h(\mathbf{x})=\text{sign}[g(\mathbf{x})]$ in the space induced by the landmarks selected, *i.e.* those with $\alpha_j\neq 0$. Note that when a ϕ^R-space (of dim. d') has been defined then the class $\mathcal{H}_{\phi^R}$ of linear classifiers learnable in this space has a finite VC-dimension ($d'+1$). Thus, according to Lemma 1 we can assess the distribution divergence in the ϕ^R-space by the empirical estimate $\hat{d}_{\mathcal{H}_{\phi^R}\Delta\mathcal{H}_{\phi^R}}$. In the following, a linear classifier learned in this framework is called a SF classifier and for sake of simplicity we will denote $\mathcal{H}_{\phi^R}$ by $\mathcal{H}$ and each similarity function is assumed to fulfill Definition 1.

IV. Domain Adaptation with Good Similarity Functions

We now present our DA method based on learning with good similarity functions. Recall that following Theorem 1, the expected target domain error is bounded by three terms: (A) the source domain error, (B) the divergence between the distributions, (C) the smallest joint error over the domains. Our idea is to minimize the expected target error by decreasing this bound. According to Balcan *et al.*, solving (2) involves to learn - only from the source domain - a good linear classifier in the explicit ϕ^R-space of similarities to a landmark set. Then, it implies a natural decreasing of (A). For minimizing (B), we want to induce a new projection space allowing to move closer the domains by selecting landmarks that are both similar to the source and target examples. To achieve this goal, we propose to add a regularization term on $\boldsymbol{\alpha}$ in (2). Due to the lack of information on the target domain, the last term (C) is hard to decrease. However, we propose to use a reverse validation approach to try to control it. In the following, we denote by $LS=\{(\mathbf{x}_i,y_i)\}_{i=1}^{d_l}$ the labeled source sample drawn from P_S ($LS_{|X}$ denoting the set $\{\mathbf{x}_i \,/\, (\mathbf{x}_i,y_i)\in LS\}_{i=1}^{d_l}$), by $R=\{\mathbf{x}'_j\}_{j=1}^{d_u}$ the landmark set and by TS the unlabeled target sample drawn from D_T.

A. Optimization Problem

Solving (2) not only minimizes the expected source error but also defines a relevant projection space for the source domain, since landmarks associated with a null weight in the solution $\boldsymbol{\alpha}$ will not be considered. According to the notion of $\mathcal{H}\Delta\mathcal{H}$-distance (Equation (1)), we propose a new additional regularizer that forces the model to provide similar outputs for pairs of source and target points, which will tend to decrease the distance between the marginal distributions. To define our regularizer, we have investigated the notion of *algorithmic robustness* proposed by Xu and Mannor [14] (see Definition 2 in Section IV-B). Their underlying idea is based on the fact that *"if a testing sample is similar to a training sample then the testing error is close to the training error"*. To ensure generalization guarantees, this framework requires that for a test point closed to a training point of the same label, the deviation between the losses of each point has to be low. They also introduce a notion of *pseudo-robustness* where it is sufficient to fulfill the robustness property for only a subpart of the training sample. This result actually assumes that the test and training data are generated from the same distribution and is thus not valid in a DA scenario. However, the authors have conjectured that it would hold for DA by taking into account a divergence measure. Despite this drawback, we propose to follow this conjecture in defining an heuristic to move closer the source and target samples. According to this idea, the samples may be similar if for pairs $(\mathbf{x}_s,\mathbf{x}_t)$ of close source and target instances of the same class, the deviation between the losses of $\mathbf{x}_s$ and $\mathbf{x}_t$ is low. This leads us to construct a term to minimize for decreasing this deviation. By considering the hinge loss of the formulation of (2), for any learned model g and any such pair $(\mathbf{x}_s,\mathbf{x}_t)$ of the class y we obtain,

$$\text{let } (a) = \big|L(g,(\mathbf{x}_s,y)) - L(g,(\mathbf{x}_t,y))\big|$$

$$(a) = \left|\left[1-y\sum_{j=1}^{d_u}\alpha_j K(\mathbf{x}_s,\mathbf{x}'_j)\right]_+ - \left[1-y\sum_{j=1}^{d_u}\alpha_j K(\mathbf{x}_t,\mathbf{x}'_j)\right]_+\right|.$$

The hinge loss is 1-lipschitz ($|[X]_+-[Y]_+|\leq|X-Y|$), then,

$$(a) \leq \left|\sum_{j=1}^{d_u}\alpha_j\big(K(\mathbf{x}_s,\mathbf{x}'_j)-K(\mathbf{x}_s,\mathbf{x}'_j)\big)\right|$$

$$(a) \leq \sum_{j=1}^{d_u} \left|\alpha_j \big(K(\mathbf{x}_s, \mathbf{x}'_j) - K(\mathbf{x}_t, \mathbf{x}'_j)\big)\right|$$
$$(a) \leq \left\|({}^t\phi^R(\mathbf{x}_s) - {}^t\phi^R(\mathbf{x}_t))\,\mathrm{diag}(\boldsymbol{\alpha})\right\|_1, \quad (3)$$

where ${}^t\phi^R(\cdot)$ is the transposed vector of $\phi^R(\cdot)$ and $\mathrm{diag}(\boldsymbol{\alpha})$ is the diagonal matrix with $\boldsymbol{\alpha}$ as main diagonal. It is hard to select the best pairs *a priori*, especially without target labels. Considering all the possible pairs is clearly intractable and we suggest, in Section V-C, a solution to build our pairs of source-target points to be moved closer. Given this pair set $\mathcal{C}_{ST} \subset LS_{|X} \times TS$, we then propose to add the new regularization term of line (3) for each pair of $\mathcal{C}_{ST}$, weighted by a parameter β. This term tends to select the landmarks with similarities close to some source and target points. Let R be a set of d_u candidate landmarks, our following global optimization (4) corresponds to (2) with the addition of our regularizer and can be easily formulated as a linear program.

$$\min_{\boldsymbol{\alpha}} \; F(\boldsymbol{\alpha}) = \frac{1}{d_l}\sum_{i=1}^{d_l}\left[1 - y_i \sum_{j=1}^{d_u}\alpha_j K(x_i, x'_j)\right]_+ + \lambda\|\boldsymbol{\alpha}\|_1 + \beta \sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} \|({}^t\phi^R(\mathbf{x}_s) - {}^t\phi^R(\mathbf{x}_t))\,\mathrm{diag}(\boldsymbol{\alpha})\|_1. \quad (4)$$

B. Sparsity Analysis and Generalization Bounds

We provide a theoretical sparsity analysis and generalization bounds of our method. We suppose (X, ρ) is a compact metric space and the similarity K is continuous in its first argument. We also make the following hypothesis on $\mathcal{C}_{ST}$,

$$\forall \mathbf{x}'_j \in R, \; \max_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} |K(\mathbf{x}_s, \mathbf{x}'_j) - K(\mathbf{x}_t, \mathbf{x}'_j)| > 0. \quad (5)$$

Hypothesis (5) means that for each coordinate $\mathbf{x}'_j$, there is at least one pair of points that brings an information with different coordinate values, which is not a strong restriction.

We begin with an analysis of the learned model sparsity.

Lemma 2. *For any $\lambda > 0$, $\beta > 0$, and any set $\mathcal{C}_{ST}$, let $B_R = \min_{\mathbf{x}'_j \in R}\big\{\max_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} |K(\mathbf{x}_s, \mathbf{x}'_j) - K(\mathbf{x}_t, \mathbf{x}'_j)|\big\}$. If $\boldsymbol{\alpha}^*$ is the optimal solution of* (4)*, then $\|\boldsymbol{\alpha}^*\|_1 \leq \dfrac{1}{\beta B_R + \lambda}$.*

Proof: See Appendix A. ∎

According to this lemma, the model sparsity depends on the parameters λ, β, and on the quantity B_R which is related to the distance between the points in $\mathcal{C}_{ST}$. When the two domains are far from each other, *i.e.* the task is hard, B_R tends to be high which can imply an increase of the sparsity.

We recall now the robustness definition and its associated theorem on the generalization ability of robust algorithms. Actually, this framework (Xu and Mannor [14]) allows us to consider the regularizers in the generalization bound.

Definition 2 ([14])**.** *Given a learning sample LS, an algorithm $\mathcal{A}$ is $(\mathbf{M}, \boldsymbol{\epsilon}(\mathbf{LS}))$* **robust** *if $X \times Y$ can be partitioned into M disjoint sets, denoted as $\{C_i\}_{i=1}^M$, such that $\forall s \in LS$,*

$$s, u \in C_i \Rightarrow \; \big|L(g, s) - L(g, u)\big| \leq \epsilon(LS),$$

with g the model learned from LS, L the loss function of $\mathcal{A}$.

Theorem 2 ([14])**.** *If $LS = \{(\mathbf{x}_i, y_i)\}_{i=1}^{d_l}$ is drawn i.i.d. from a distribution P and if the algorithm $\mathcal{A}$ is $(M, \epsilon(LS))$ robust, then for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\mathrm{err}_P(\mathcal{A}_{LS}) \; \leq \hat{\mathrm{err}}_P(\mathcal{A}_{LS}) + \epsilon(LS) + L^{UP}\sqrt{\frac{2M\ln 2 + 2\ln(1/\delta)}{d_l}},$$

where err_P and $\hat{\mathrm{err}}_P$ are respectively the expected and the empirical errors over P, L being upper bounded by L^{UP}.

We can prove that our method is robust on the source domain which leads to the following generalization bound for the expected source error.

Theorem 3. *If $LS = \{(\mathbf{x}_i, y_i)\}_{i=1}^{d_l}$ is drawn i.i.d. from P_S, then* (4) *is $(\mathbf{2M}_\eta, \frac{\mathbf{N}_\eta}{\beta\mathbf{B_R}+\boldsymbol{\lambda}})$* **robust on the source domain** $\mathbf{P_S}$*, where $N_\eta = \max_{\substack{\mathbf{x}_a,\mathbf{x}_b\sim D_S \\ \rho(\mathbf{x}_a,\mathbf{x}_b)\leq\eta}} \|{}^t\phi^R(\mathbf{x}_a) - {}^t\phi^R(\mathbf{x}_b)\|_\infty$ with $\eta > 0$ and M_η is the η-covering number of X (see [14] for more details). Thus for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\mathrm{err}_S(h) \; \leq \hat{\mathrm{err}}_S(h) + \frac{N_\eta}{\beta B_R + \lambda} + \sqrt{\frac{4M_\eta \ln 2 + 2\ln\frac{1}{\delta}}{d_l}}.$$

Proof: See Appendix B. ∎

We now derive a generalization bound for the expected target domain error directly from Theorems 1 and 3.

Theorem 4. *If $LS = \{(\mathbf{x}_i, y_i)\}_{i=1}^{d_l}$ is drawn i.i.d. from the source domain P_S, for every h in the hypothesis class $\mathcal{H}$ of SF classifier, for any $\delta > 0$, with probability at least $1 - \delta$,*

$$\mathrm{err}_T(h) \leq \hat{\mathrm{err}}_S(h) + \frac{N_\eta}{\beta B_R + \lambda} + \sqrt{\frac{4M_\eta \ln 2 + 2\ln\frac{1}{\delta}}{d_l}} + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_T) + \nu,$$

where ν is the joint error over the domains, $d_{\mathcal{H}\Delta\mathcal{H}}(D_S, D_T)$ is the $\mathcal{H}\Delta\mathcal{H}$-distance between the marginal distribution.

The constant $\frac{N_\eta}{\beta B_R + \lambda}$ depends on our regularizers and on N_η that can be obtained as small as wished by continuity of K.

C. Reverse Classifier and Validation

A crucial point is the choice of the hyperparameters of our method. We propose to follow the idea of *TrCV* from Zhong *et al.* [15]. However, this approach relies on valid kernels and some few target labels. Since we may consider non-PSD and non-symmetric similarity functions and no label on the target domain, we make a little adaptation. We use their concept of *reverse validation* based on a *reverse classifier* evaluated on the source domain (Fig. 2), but directly in

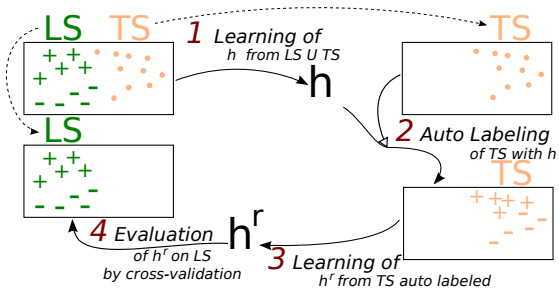


Figure 2. The reverse validation. Step **1**: *Learning* h with (4). Step **2**: *Auto-labeling* the target sample by h. Step **3**: *Learning* h^r on the auto-labeled target sample with (2). Step **4**: *Evaluation* of h^r on LS.

the projection space found. The justification of this choice comes from the fact that if the domains are sufficiently close and related, then the reverse classifier must be also efficient on the source task [10]. In other words, in the projection space it is possible to pass from one problem to another. Since we do not have any information on the target distribution we define the reverse classifier h^r as the best SF classifier learned with (2), in the ϕ^R-space, from the target sample $\{(\mathbf{x},h(\mathbf{x}))\}_{\mathbf{x}\in TS}$ self-labeled by the classifier h learned with (4). Given k-folds on the source sample ($LS = \cup_{i=1}^{k} LS_i$), we use $k-1$ labeled folds as labeled examples for learning h and we evaluate h^r on the last k^{th} fold. The error corresponds to the mean of the error over the k-folds: $\hat{\text{err}}_S(h^r) = \frac{1}{k}\sum_{i=1}^{k} \hat{\text{err}}_{LS_i}(h^r)$. We finally consider the DA capability of the classifier as an empirical estimation of the last term ν of the bound of Theorem 1 defined by $\hat{\nu} = \hat{\text{err}}_S(h^r) + \hat{\text{err}}_T(h^r)$, where $\hat{\text{err}}_T(h^r)$ being evaluated by cross-validation over the auto-labeled target sample. We then select the hyperparameters minimizing $\hat{\nu}$. Note that with this choice and the minimization of (4), we try to minimize the three terms of the bound of Theorem 1.

V. An Iterative Reweighting: a Way to Lighten the Search of the Projection Space

Building the pair set $\mathcal{C}_{ST}$ is *a priori* hard since we have no target label. Moreover, the set of relevant pairs allowing a good adaptation depends generally on the considered task and testing all the possible sets, with the reverse validation, is clearly intractable. To tackle this problem, we propose an iterative approach based on a selection of a limited number of pairs and a reweighting scheme of the similarities keeping close distributions. We finally present a stopping criterion based on the empirical estimation of the joint error.

A. Selecting the pairs of $\mathcal{C}_{ST}$

We propose to construct pairs from two subsets of the two samples $U_S \subseteq LS_{|X}$ and $U_T \subseteq TS$ of equal size. We select them, at a given iteration l, according to the *reverse model* g^r_{l-1} associated with the reverse classifier h^r_{l-1} computed in the previous iteration. They correspond to the examples on which this model is highly or weakly confident on the labels. Let $\delta_S^H, \delta_T^H, \delta_S^L, \delta_T^L$ a set of positive parameters, U_S and U_T are defined as follows such that $|U_S| = |U_T| \leq N$,

$$\begin{cases} U_S = \left\{\mathbf{x} \in LS_{|X} \ : \ |g^r_l(\mathbf{x})| > \delta_S^H \text{ OR } |g^r_l(\mathbf{x})| < \delta_S^L\right\} \\ U_T = \left\{\mathbf{x} \in TS \ \ : \ |g^r_l(\mathbf{x})| > \delta_T^H \text{ OR } |g^r_l(\mathbf{x})| < \delta_T^L\right\}. \end{cases}$$

Using these two sets, we build $\mathcal{C}_{ST} \subset U_S \times U_T$ as a bipartite matching by minimizing the euclidean distance in the ϕ^R_l-space: $\sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} \left\| {}^t\phi^R_l(\mathbf{x}_s) - {}^t\phi^R_l(\mathbf{x}_t)\right\|_2^2$. This is a way to infer pairs of close source-target points in the ϕ^R_l-space considered at iteration l. Limiting the subsets to small[1] N allows us to build efficiently this bipartite matching, which is not a too restrictive heuristic since the notion of pseudo-robustness does not require to consider all the points.

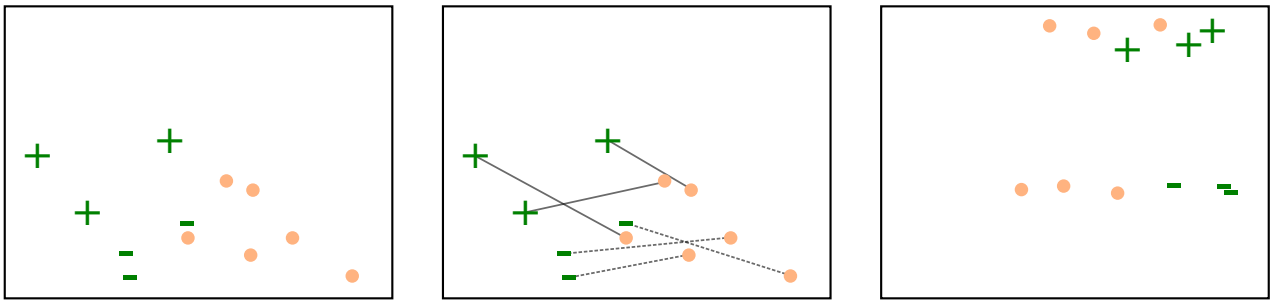
B. A New Projection Space by Iterative Reweighting

The landmarks selected by solving (4), *i.e.* those with non null α_j, define a projection space where the distributions tend to be close. We propose to re-use these α_j to force the new space to move closer the distributions by reweighting the similarity function according to $\boldsymbol{\alpha}$. Suppose at a given iteration l, with a similarity function K_l, we obtain new weights $\boldsymbol{\alpha}^l$. Then, we propose to define K_{l+1} by weighting K_l conditionally to each landmark of R such that, $\forall \mathbf{x}'_j \in R$, $K_{l+1}(\mathbf{x},\mathbf{x}'_j) = \alpha^l_j K_l(\mathbf{x},\mathbf{x}'_j)$ (eventually normalized to ensure $K_{l+1} \in [-1,1]$). It can be seen as a kind of contraction of the space to keep the $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ low. Indeed, in this new ϕ^R_{l+1}-space defined by K_{l+1}, the points of each pair of $\mathcal{C}_{ST}$ are naturally close since, by construction, our regularizer corresponds exactly to minimize their L_1-distance in the ϕ^R_{l+1}-space. Actually, we have, $\forall (\mathbf{x}_s,\mathbf{x}_t) \in \mathcal{C}_{ST}$, $\|{}^t\phi^R_{l+1}(\mathbf{x}_s) - {}^t\phi^R_{l+1}(\mathbf{x}_t)\|_1 = \|{}^t(\phi^R_l(\mathbf{x}_s) - {}^t\phi^R_l(\mathbf{x}_t))\text{diag}(\boldsymbol{\alpha}^l)\|_1$. An illustration of this procedure is provided on Fig. 3. We then iterate the process in the new ϕ^R_{l+1}-space. The possible reweightings are related to the different hyperparameters $\delta^{H/L}_{S/T}$ (linked to $\mathcal{C}_{ST}$) and λ, β of (4) that are selected according to reverse validation. Recall that, since we are not interested in using valid kernels, we do not have to keep any notion of symmetry or positive semi-definiteness for K_{l+1}. However, our normalization remains valid only if the new similarity function is still good on the source domain, which can be empirically estimated by evaluating ϵ and γ from Definition 1 on LS. In fact, we pay attention to keep only those that offer the best (ϵ,γ)-guarantees, ensuring a sufficiently good similarity. Note that a bad similarity would lead to a dramatic increase of the expected source error.

C. Stopping Criterion

We consider here the estimated joint error $\hat{\nu}$ related to the adaptation capability in the current space. Controlling this term and its decreasing during the iterative process can provide a nice way to stop the algorithm. Following

[1] In our experiments we take $N \leq 30$.



(a) The points in a ϕ_l^R-space, $|R|=2$.

(b) The associated bipartite matching $\mathcal{C}_{ST}$: a line corresponds to a pair $(\mathbf{x}_s, \mathbf{x}_t)$.

(c) The points in the new ϕ_{l+1}^R-space after the reweighting procedure.

Figure 3. An iteration of the reweighting process. The source points are (dark) green (pos. $+$, neg. $-$), the unlabeled target ones are (light) orange circle.

Algorithm 1 DASF: Domain Adaptation with Similarity Function

input similarity function K, set R, samples LS and TS

output classifier h_{DASF}

$h_0(\cdot) \leftarrow \text{sign}\left[\frac{1}{|R|}\sum_{j=1}^{|R|} K(\cdot, \mathbf{x}'_j)\right]$; $K_1 \leftarrow K$; $l \leftarrow 1$

while The stopping criterion is not verified **do**

Select $U_S \subseteq LS_{|X}$, $U_T \subseteq TS$ with h_{l-1}^r ; Build $\mathcal{C}_{ST}$

$\boldsymbol{\alpha}^l \leftarrow$ Solve our Problem with K_l and $\mathcal{C}_{ST}$

$K_{l+1} \leftarrow$ Update K_l according to $\boldsymbol{\alpha}^l$

Update R ; $l++$

end while

return $h_{DASF}(\cdot) = \text{sign}\left[\sum_{x'_j \in R} \alpha_j^l K_l(\cdot, \mathbf{x}'_j)\right]$

Section IV-C, at a given iteration l this term is defined by $\hat{\nu}_l = \hat{\text{err}}_S(h_l^r) + \hat{\text{err}}_T(h_l^r)$. An increasing of $\hat{\nu}_l$ between two iterations means that the new projection space found is no longer relevant and the current one must be preferred. Then, our process stops at iteration l when $\hat{\nu}_{l+1}$ has reached a convergence point or has increased significantly. This criterion allows us to ensure the algorithm stops since the joint error is positive and bounded by 0. The global iterative algorithm (named DASF) is described in Algorithm 1.

VI. Experiments

In this section, we evaluate our approach DASF on a synthetic toy problem and on a real image annotation task. The similarity function used is based on a Gaussian kernel. To obtain a non-symmetric and non-PSD similarity K^*, we apply the following normalization from a Gaussian kernel K: given a set of landmarks R, for every $\mathbf{x}'_j \in R$,

$$K^*(., \mathbf{x}'_j) = \begin{cases} \frac{K(., \mathbf{x}'_j) - \mu_{\mathbf{x}'_j}}{\sigma_{\mathbf{x}'_j}} & \text{if } -1 \leq \frac{K(., \mathbf{x}'_j) - \hat{\mu}_{\mathbf{x}'_j}}{\hat{\sigma}_{\mathbf{x}'_j}} \leq 1, \\ -1 & \text{if } \frac{K(., \mathbf{x}'_j) - \hat{\mu}_{\mathbf{x}'_j}}{\hat{\sigma}_{\mathbf{x}'_j}} \leq -1, \\ 1 & \text{if } \frac{K(., \mathbf{x}'_j) - \hat{\mu}_{\mathbf{x}'_j}}{\hat{\sigma}_{\mathbf{x}'_j}} \geq 1, \end{cases}$$

where $\hat{\mu}_{\mathbf{x}'_j}$ is the empirical mean of similarities to $\mathbf{x}'_j$ over $LS_{|X} \cup TS$ and $\hat{\sigma}_{\mathbf{x}'_j}$ is the empirical unbiased estimate of the standard deviation. However, depending on the considered samples, K^* does not always offer better (ϵ,γ,τ)-good guarantees than the Gaussian kernel. In the following, we only indicate the similarity which obtains the best results. Problems favouring K^* are indicated with a *, as we will see they correspond generally to harder DA tasks. We compare DASF with a classical SVM learned only on the source domain, the semi-supervised Transductive SVM [16] (TSVM) and the DA method DASVM [10]. We use a classical Gaussian kernel for these three methods to facilitate the comparison. We use the SVM-light library [17] with parameters tuned by cross-validation on the source data for SVM and TSVM. DASVM is implemented with the LibSVM library [18]. Parameters of DASVM and DASF are tuned according to a grid search. We also measure the behavior of a SF classifier trained only on the source domain. For DASF and SF, the landmarks are taken from the labeled source sample. Following (1), we assess the distance $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ between the marginal distributions by learning a SF classifier to separate source from target samples, a small value indicates close distributions while a larger value indicates a hard DA task.

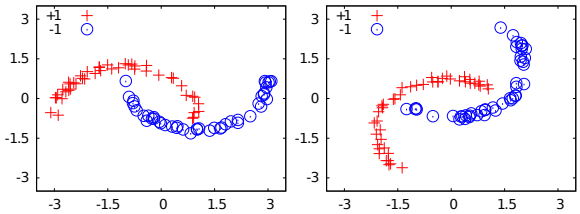


Figure 4. Left: a source sample. Right: a target sample with a 50° rotation.

A. Synthetic Toy Problem

As the source domain we consider a classical binary problem with two intertwinning moons, each class corresponding to one moon (Fig. 4). We then consider 8 different target domains by rotating anticlockwise the source domain according to 8 angles. The higher the angle is, the more difficult the problem becomes. For each domain, we generate

Table I
THE RESULTS (AVERAGE ACCURACY) OBTAINED FOR THE TWO MOONS TOY PROBLEM.

ROTATION ANGLE	20°	30°	40°	50°	60°*	70°*	80°*	90°*
SVM	89.68 ±0.78	75.99 ±0.92	68.84 ±0.85	60.00 ±1.08	47.18 ±2.82	26.12 ±3.12	19.22 ±0.28	17.2 ±0.37
SV	18 ±0.99							
SF	92.4 ±3.13	81.81 ±4.62	72.55 ±7.60	57.85 ±4.81	43.93 ±4.46	39.2 ±9.64	35.93 ±10.93	36.73 ±10.17
LAND.	24 ±1.72				22 ±3.57	20 ±2.06	20 ±2.82	20 ±1.51
TSVM	**100** ±0.00	78.98 ±2.31	74.66 ±2.17	70.91 ±0.88	64.72 ±9.10	21.28 ±1.26	18.92 ±1.10	17.49 ±1.12
SV	28 ±1.92	37 ±3.77	37 ±2.66	37 ±1.50	38 ±2.67	35 ±2.93	37 ±2.10	36 ±1.69
DASVM	100 ±0	78.41 ±4.56	71.63 ±4.16	66.59 ±4.01	61.57 ±4.15	25.34 ±3.28	21.07 ±2.33	18.06 ±2.66
SV	20 ±3.13	20 ±4.42	26 ±6.80	28 ±2.81	29 ±3.62	34 ±7.58	38 ±6.20	23 ±4.95
DASF	99.80 ±0.40	**99.55** ±1.19	**91.03** ±3.30	**81.27** ±4.36	**65.23** ±6.36	**61.95** ±4.88	**60.91** ±2.24	**59.75** ±2.11
LAND.	**10** ±2.32	**10** ±1.59	**9** ±2.21	**8** ±3.27	**4** ±0.99	**4** ±2.16	**4** ±1.84	**3** ±1.06
$\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ IN ϕ_0^R	0.58 ±0.04	1.16 ±0.04	1.31 ±0.04	1.34 ±0.04	1.34 ±0.03	1.32 ±0.03	1.33 ±0.03	1.31 ±0.05
$\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ IN ϕ_{final}^R	0.33 ±0.12	0.66 ±0.11	0.82 ±0.13	0.85 ±0.11	0.39 ±0.15	0.40 ±0.05	0.49 ±0.12	0.45 ±0.09

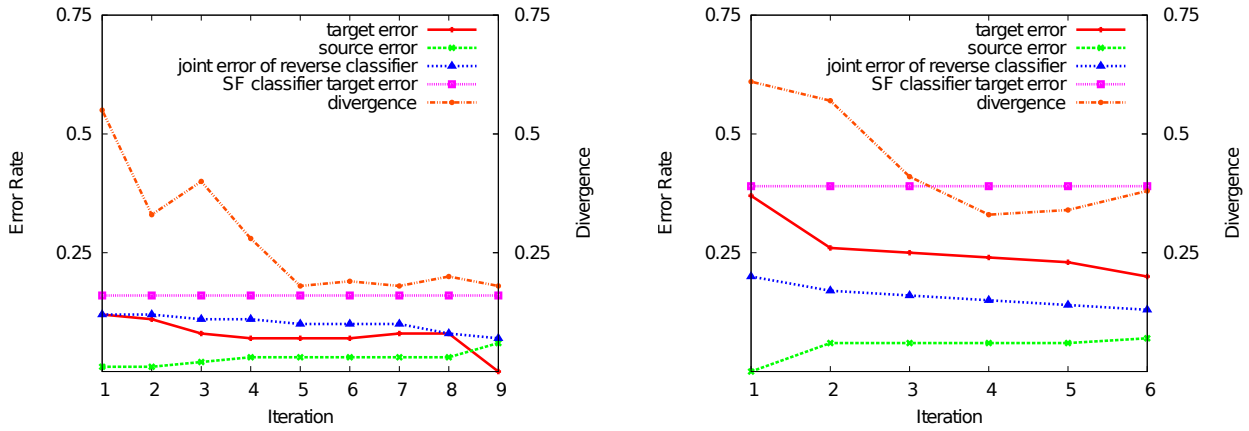


Figure 5. Two DASF executions. On the left with a 30° rotation, on the right with a 50° rotation. On the left y-axis is the error range, on the right y-axis the divergence range. We provide the error rates of the classifiers h_l built at each iteration on the source and target test samples, the divergence $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ between the distributions, the joint error of the reverse classifier, the error on the target test sample of a SF classifier learning without DA as a baseline.

300 instances (150 of each class). Moreover, to assess the generalization ability of our approach, we evaluate each algorithm on an independent test set of 1500 points drawn from the target domain. Each DA problem is repeated 10 times. The average accuracy of each method is reported on Tab. I. We also indicate the average number of support vectors (SV) used by SVM, TSVM and DASVM, the number of landmarks (LAND.) selected by SF and DASF and an estimation of $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ in the initial ϕ_0^R-space and the final ϕ_{final}^R-space. We can make the following remarks.

• DASF outperforms in average the other methods. It is significantly better for every task of angle greater than 20°. While the accuracy of TSVM and DASVM falls down from 60°, DASF remains still competitive even when the difficulty increases. In this case, the normalized K^* is preferred.

• The landmark number is significantly lower than the SV number, which confirms that DASF produces very sparse models with good performances. The gain ratio is between 3 to 12. The DASF classifiers are also sparser than the SF ones which use a L1-regularization too. Finally, they tend to be sparser for difficult problems as suggested by Lemma 2.

• The distance between the domains is lower at the last iteration - between 1 and 9 - showing our iterative approach is effectively able to quickly move closer the distributions.

Table II
THE RESULTS ON THE PASCALVOC TEST TARGET DOMAINS ACCORDING TO THE F-MEASURE. AVG. IS THE AVERAGE RESULT.

CONCEPT	BIRD	BOAT	BOTTLE*	BUS	CAR	CAT	CHAIR	CYCLE	COW	DININGTABLE
SVM	0.18	0.29	0.01	0.16	0.28	**0.23**	0.24	0.10	0.15	0.15
SV	867	351	587	476	1096	882	1195	392	681	534
SF	0.18	0.27	0.11	0.12	0.34	0.20	0.21	0.10	0.11	0.10
LAND.	237	203	233	212	185	**178**	241	139	239	**253**
TSVM	0.14	0.14	0.11	0.16	0.37	0.14	0.22	0.13	0.12	0.13
SV	814	704	718	445	631	779	864	390	888	515
DASVM	0.16	0.22	0.11	0.14	0.37	0.20	0.23	0.14	0.11	0.15
SV	922	223	295	421	866	1011	1418	706	335	536
DASF	**0.20**	**0.32**	**0.12**	**0.17**	**0.38**	**0.23**	**0.26**	**0.16**	**0.16**	**0.16**
LAND.	**50**	**184**	**78**	**94**	**51**	378	**229**	**192**	**203**	372

CONC.	DOG*	HORSE	MONITOR	MOTORBIKE	PERSON*	PLANE	PLANT	SHEEP	SOFA	TRAIN
SVM	0.24	0.31	**0.16**	0.17	0.56	0.34	0.12	0.16	0.16	0.36
SV	436	761	698	670	951	428	428	261	631	510
SF	0.18	0.24	0.12	0.17	0.46	0.34	0.13	0.12	0.13	0.20
LAND.	200	**247**	**203**	243	226	**178**	**236**	**128**	224	202
TSVM	0.22	0.17	0.12	0.12	0.44	0.18	0.10	0.12	0.15	0.19
SV	704	828	861	861	1111	585	406	474	866	652
DASVM	0.22	0.23	0.12	0.14	0.55	0.30	0.12	0.13	0.17	0.28
SV	**180**	802	668	841	303	356	1434	246	486	407
DASF	**0.25**	**0.32**	**0.16**	**0.18**	**0.58**	**0.35**	**0.15**	**0.20**	**0.18**	**0.42**
LAND.	391	384	287	**239**	**6**	181	293	153	**167**	**75**

CONC.	AVG.
SVM	0.22
SV	642
SF	0.19
LAND.	210
TSVM	0.17
SV	705
DASVM	0.20
SV	622
DASF	**0.25**
LAND.	**200**

As evoked before, DASF tends to build a small projection space for hard problems, probably to have sufficiently close domains, but this may imply a loss of expressiveness.
Fig. 5 shows two DASF executions on two DA problems. For these cases, the $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ distance decreases significantly in comparison with iteration 1. DASF stops when the empirical joint error reaches a minimum after decreasing continuously. Note that the final projection space found is not always the one with the lowest $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$, this is because we need to find a compromise between the minimization of $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ and the one of the source error. Thanks to the iterative procedure, DASF is then able to slightly autocorrect the space found when it allows a better adaptation. For the $30°$ example, DASF finds a null error classifier on the target test sample. For the more difficult $50°$ example, DASF finds a classifier, performing better than the SF classifier learned only on the source data. Note that the source error increases, which is expected since we aim at being performing on the target domain.

B. Image Classification

In this section, we evaluate DASF on PascalVOC'07 [19] and TrecVid'07 [20] corpora. The goal is to identify visual objects in images and videos. TrecVid corpus is constituted of images extracted from videos and can be seen as an image corpus. Visual features used for those experiments are based on the prediction scores of 15 "intermediate" visual concepts (ANIMAL, BUILDING, CAR, CARTOON, EXPLOSION-FIRE, FLAG-US, GREENERY, MAPS, ROAD, SEA, SKIN_FACE, SKY, SNOW, SPORTS, STUDIO_SETTING) which have been successfully used in previous TrecVid evaluations. Each of those intermediate concepts are detected using SVM classifiers

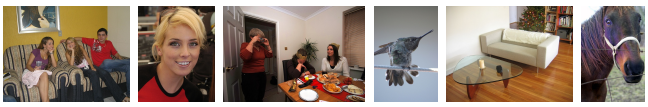


Figure 6. PascalVOC: The 6 landmarks selected for the concept PERSON, the first 3 images are positive and the last 3 are negative.

from color moments and edge orientations on 260 blocs of 32×32 pixels (data dimension is 3900) according to [21]. We made two experiments.

First, the PascalVOC benchmark is constituted of a set of 5000 training images, a set of 5000 test images and a list of 20 concepts to identify. Training and test sets are in fact relatively close ($\hat{d}_{\mathcal{H}\Delta\mathcal{H}} \simeq 0.05$) and a DA step is not necessary. We rather propose to evaluate the DA capability of our algorithm when the ratio $+/-$ is different between the source and target samples, leading to an harder DA task. Our objective is not to provide a solution in such a situation (specific methods exist [22]), but rather to evaluate if our method can avoid negative transfer and improve the accuracy over the test set. In general, the ratio between positive and negative examples (ratio $+/-$) is less than 10% in this dataset. For each concept, we generated a source sample constituted of all the training positive data and the negatives data are independently drawn such that the ratio $+/-$ is $\frac{1}{3}/\frac{2}{3}$. We keep the original test set as the target sample. We applied the 5 methods previously described for learning a binary classifier for each concept. Due to the relatively small ratio $+/-$ in the target sample, we evaluate the performances according to the well known F-measure. The results are

Table III
THE RESULTS OBTAINED ON THE TRECVID TARGET DOMAINS ACCORDING TO THE F-MEASURE. AVG. IS THE AVERAGE RESULT.

CONCEPT	BOAT*	BUS*	CAR*	MONITOR*	PERSON*	PLANE*	AVG.
SVM	0.56	0.25	0.43	0.19	0.52	0.32	0.38
SV	351	476	1096	698	951	428	667
SF	0.49	0.46	0.50	0.34	0.45	0.54	0.46
LAND.	214	224	**176**	246	226	178	211
TSVM	0.56	0.48	0.52	0.37	0.46	0.61	0.50
SV	498	535	631	741	1024	259	615
DASVM	0.52	0.46	**0.55**	0.30	0.54	0.52	0.48
SV	202	222	627	523	274	450	383
DASF	**0.57**	**0.49**	**0.55**	**0.42**	**0.57**	**0.66**	**0.54**
LAND.	**120**	**130**	254	**151**	**19**	**7**	**113**

reported on Tab. II. First, TSVM and DASVM perform badly, probably because of the difference between target and source ratios $+/-$ which cannot be estimated due to the lack of information on the target sample. SVM performs often better than the two previous ones which can be explained by the similarity between the train and test data. DASF has the best behavior in average. It always improves the results of a SF classifier, avoiding negative transfer, and is the best for 18 concepts. Moreover, it always outputs significantly sparser models. As an illustration, we give on Fig. 6 the landmarks selected for the concept PERSON.

In the last experiment, we selected the 6 common concepts between TrecVid'07 and PascalVOC'07. For each concept, we keep our PascalVOC training set as the source domain and take, as the target domain, a TrecVid set of examples with the same ratio $+/-$ as the training set. $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}$ is about 1.4 justifying the high difference between the two corpora and thus a potentially hard DA task. The results evaluated with the F-measure are reported on Tab. III. DASF obtains the best results in average and outputs again significantly sparser models. Finally, for those hard tasks the normalized similarity K^* is always preferred, showing that DASF is effectively able to deal with non-symmetric non-PSD good similarities. K^* has the interest of incorporating some target information which seems useful for hard DA tasks.

VII. CONCLUSION

In this paper, we have proposed a novel domain adaptation approach that takes advantage of the framework of Balcan *et al.* [11], [12] allowing to deal with similarity functions potentially non-PSD and non-symmetric. The method relies on a regularization term that helps to build a projection space, made of similarities to landmark points, by selecting those both close to the source and target examples. The linear formulation of the method enables the proposed algorithm to output sparse models (even when the DA task is hard). We have also studied the generalization ability of our method according to the framework of robustness allowing us to take into account our regularizers. Moreover, we have proposed an effective iterative process to lighten the search of the projection space by reweighting the similarities. We have experimentally shown good adaptation abilities on various tasks and our method always outputs sparser models which is clearly an advantage for a large scale application perspective.

As a future work, we intend to extend DASF to allow the use of a small labeled target sample to help a better projection space construction, potentially augmented as in [23]. Another goal would be to exploit multimodality [24].

ACKNOWLEDGMENT

This work was supported in part by the french project VideoSense ANR-09-CORD-026 of the ANR in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886.

REFERENCES

[1] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[2] J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. Lawrence, *Dataset Shift in Machine Learning*. MIT Press, 2009.

[3] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Vaughan, "A theory of learning from different domains," *Machine Learning Journal*, vol. 79, no. 1-2, pp. 151–175, 2010.

[4] A. Bergamo and L. Torresani, "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach," in *Proceedings of NIPS*, 2010.

[5] H. Daumé III, "Frustratingly easy domain adaptation," in *Proceedings of ACL*, 2007.

[6] G. Schweikert, C. Widmer, B. Schölkopf, and G. Rätsch, "An empirical analysis of domain adaptation algorithms for genomic sequence analysis," in *Proceedings of NIPS*, 2008, pp. 1433–1440.

[7] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," in *Proceedings of COLT*, 2009, pp. 19–30.

[8] J. Huang, A. Smola, A. Gretton, K. Borgwardt, and B. Schölkopf, "Correcting sample selection bias by unlabeled data," in *Proceedings of NIPS*, 2006, pp. 601–608.

[9] M. Sugiyama, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *Proceedings of NIPS*, 2007.

[10] L. Bruzzone and M. Marconcini, "Domain adaptation problems: A DASVM classification technique and a circular validation strategy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 770–787, 2010.

[11] M.-F. Balcan, A. Blum, and N. Srebro, "Improved guarantees for learning via similarity functions," in *Proceedings of COLT*, 2008, pp. 287–298.

[12] ——, "A theory of learning with similarity functions," *Machine Learning J.*, vol. 72, no. 1-2, pp. 89–112, 2008.

[13] S. Ben-David, T. Lu, T. Luu, and D. Pal, "Impossibility theorems for domain adaptation," *JMLR W&CP*, vol. 9, pp. 129–136, 2010.

[14] H. Xu and S. Mannor, "Robustness and generalization," in *Proceedings of COLT*, 2010, pp. 503–515.

[15] E. Zhong, W. Fan, Q. Yang, O. Verscheure, and J. Ren, "Cross validation framework to choose amongst models and datasets for transfer learning," in *Proceedings of ECML-PKDD*, ser. LNCS, vol. 6323. Springer, 2010, pp. 547–562.

[16] V. Vapnik, *Statistical Learning Theory*. Springer, 1998.

[17] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proceedings of ICML*, 1999, pp. 200–209.

[18] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, www.csie.ntu.edu.tw/~cjlin/libsvm.

[19] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," www.pascal-network.org/challenges/VOC/voc2007/workshop/, 2007.

[20] A. Smeaton, P. Over, and W. Kraaij, "High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements," in *Multimedia Content Analysis, Theory and Applications*. Springer Verlag, 2009, pp. 151–174.

[21] S. Ayache, G. Quénot, and J. Gensel, "Image and video indexing using networks of operators," *Journal on Image and Video Processing*, vol. 2007, pp. 1:1–1:13, 2007.

[22] C. Seah, I. Tsang, Y. Ong, and K. Lee, "Predictive distribution matching svm for multi-domain learning," in *Proceedings of ECML PKDD*, ser. LNCS, vol. 6321. Springer, 2010, pp. 231–247.

[23] H. Daumé III, A. Kumar, and A. Saha, "Co-regularization based semi-supervised domain adaptation," in *Proceedings of NIPS*, 2010.

[24] L. Duan, I. Tsang, D. Xu, and T. Chua, "Domain adaptation from multiple sources via auxiliary classifiers," in *Proceedings of ICML*, 2009, p. 37.

APPENDIX

A. Proof of Lemma 2

Proof: Recall $F(.)$ refers to (4). First, for any $\boldsymbol{\alpha}$,

$$\text{let } (b) = \sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} \|({}^t\phi^R(\mathbf{x}_s) - {}^t\phi^R(\mathbf{x}_t))\,\mathrm{diag}(\boldsymbol{\alpha})\|_1$$

$$(b) = \sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} \sum_{j=1}^{d_u} \left|\alpha_j \left(K(\mathbf{x}_s,\mathbf{x}'_j) - K(\mathbf{x}_t,\mathbf{x}'_j)\right)\right|$$

$$(b) = \sum_{j=1}^{d_u} \left[|\alpha_j| \left(\sum_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} |K(\mathbf{x}_s,\mathbf{x}'_j) - K(\mathbf{x}_t,\mathbf{x}'_j)| \right)\right]$$

$$(b) \geq \sum_{j=1}^{d_u} \left[|\alpha_j| \max_{(\mathbf{x}_s,\mathbf{x}_t)\in\mathcal{C}_{ST}} |K(\mathbf{x}_s,\mathbf{x}'_j) - K(\mathbf{x}_t,\mathbf{x}'_j)|\right].$$

From Hypothesis (5) $B_R > 0$, thus $(b) \geq \|\boldsymbol{\alpha}\|_1 B_R$. Then,

$$\|\boldsymbol{\alpha}^*\|_1(\lambda+\beta B_R)+\frac{1}{d_l}\sum_{i=1}^{d_l}\left[1-\sum_{j=1}^{d_u}\alpha_j^* K(\mathbf{x}_i,\mathbf{x}'_j)\right]_+ \leq F(\boldsymbol{\alpha}^*).$$

Since $\boldsymbol{\alpha}^*$ is optimal, we have $F(\boldsymbol{\alpha}^*)\leq F(\mathbf{0})=1$, with $\mathbf{0}$ the null vector. Then, we obtain directly, $\|\boldsymbol{\alpha}^*\|_1 \leq \frac{1}{\beta B_R+\lambda}$. ■

B. Proof of Theorem 4

Proof: Let (X,ρ) a compact metric space. Then, with $\eta>0$ and by definition of covering number, we can partition X in M_η(finite) subsets, s.t. for $\mathbf{x}_1,\mathbf{x}_2$ belonging to the same subset $\rho(\mathbf{x}_1,\mathbf{x}_2)\leq\eta$. With Y divided in 2 subsets $\{\{-1\},\{+1\}\}$ and following [14], we partition $X\times Y$ in $2M_\eta$ subsets such that points in the same subset are of the same class. Given K a continuous similarity in its first argument, a training set $LS=\{(\mathbf{x}_i,y_i)\}_{i=1}^{d_l}$, a landmark set $R=\{\mathbf{x}'_j\}_{j=1}^{d_u}$, $\lambda>0$, $\beta>0$ and a set $\mathcal{C}_{ST}$, let $\boldsymbol{\alpha}^*$ be the optimal solution of (4). For any $s_1=(\mathbf{x}_1,y_1)\in LS$, any $s_2=(\mathbf{x}_2,y_2)$ s.t. s_1, s_2 belong to the same subset, thus $y_1=y_2$ and $\rho(\mathbf{x}_1,\mathbf{x}_2)\leq\eta$ then,

$$\text{let } (a) = \left|L\left(g,(\mathbf{x}_1,y)\right) - L\left(g,(\mathbf{x}_2,y)\right)\right|$$

$$(a) = \left| \left[1-y_1\sum_{j=1}^{d_u}\alpha_j^* K(\mathbf{x}_1,\mathbf{x}'_j)\right]_+ - \left[1-y_1\sum_{j=1}^{d_u}\alpha_j^* K(\mathbf{x}_2,\mathbf{x}'_j)\right]_+ \right|.$$

By 1-lipschitz property of the hinge loss, the successive application of Holder inequality and Lemma 2, we have,

$$(a) \leq \|\boldsymbol{\alpha}^*\|_1 \|{}^t\phi^R(\mathbf{x}_1) - {}^t\phi^R(\mathbf{x}_2)\|_\infty$$

$$(a) \leq \|\boldsymbol{\alpha}^*\|_1 \max_{\substack{\mathbf{x}_a,\mathbf{x}_b\sim D_S \\ \rho(\mathbf{x}_a,\mathbf{x}_b)\leq\eta}} \left\{\|{}^t\phi^R(\mathbf{x}_a) - {}^t\phi^R(\mathbf{x}_b)\|_\infty\right\} \leq \frac{N_\eta}{\beta B_R+\lambda},$$

with $N_\eta = \max_{\substack{\mathbf{x}_a,\mathbf{x}_b\sim D_S \\ \rho(\mathbf{x}_a,\mathbf{x}_b)\leq\eta}} \left\{\|{}^t\phi(\mathbf{x}_a)^R - {}^t\phi(\mathbf{x}_b)^R\|_\infty\right\}$ which is finite by continuity of K in its first argument and definition of covering number. Then, the algorithm associated to (4) is $\left(M_\eta, \frac{N_\eta}{\beta B_R+\lambda}\right)$ robust. Since the upper bound of hinge loss $\left[1-y\sum_{j=1}^{d_u}\alpha_j K(.,\mathbf{x}'_j)\right]_+$ is 1, we directly derive the bound of Theorem 3 from Theorem 2. ■